

# Claims

- [c1] 1. A method of accessing data contained in a first file, wherein said first file is comprised in a plurality of files stored on a secondary storage, said secondary storage comprising a plurality of clusters, a file allocation table (FAT) indicating a corresponding set of clusters allocated to each of said plurality of files, said method comprising: determining a set of identifiers, wherein each of said set of identifiers identifies a corresponding one of said set of clusters allocated to said first file; and storing said set of identifiers in a random access memory (RAM), wherein storing said set of identifiers in said RAM enables faster access to data related to said first file, and not having to store said entire FAT in said RAM minimizes memory space requirements in said RAM.
- [c2] 2. The method of claim 1, wherein each of said plurality of clusters is identified by a corresponding one of a plurality of identifiers, said FAT storing each of said sets of identifiers in the form of a linked list, wherein an order specified by said linked list indicates the sequence in which said set of clusters are used to store data con-

tained in said first file, said method comprising:  
traversing said linked list to retrieve said set of identifiers in said order, wherein said storing stores said set of identifiers in said RAM.

- [c3] 3. The method of claim 2, wherein said set of identifiers are stored according to a technique by which each of said set of identifiers can be retrieved with fewer instructions than the number of instructions required to access the same identifier from said FAT in said secondary storage.
- [c4] 4. The method of claim 2, wherein said set of identifiers are stored in the form of an array from which each identifier can be retrieved by a single access.
- [c5] 5. The method of claim 4, further comprising:  
receiving a start offset of data to be accessed;  
computing a cluster index by dividing said start offset by a number of bytes in each of said plurality of clusters;  
and  
accessing said array using said cluster index to determine a specific one of said set of identifiers, wherein said data to be accessed is present in a cluster identified by said specific one of said set of identifiers.
- [c6] 6. The method of claim 5, wherein data stored in said file

represents a song.

- [c7] 7. A method of implementing an application in a system containing a small memory, wherein said system supports a file system on a secondary storage, wherein said secondary storage comprises a plurality of clusters, wherein said file system comprises a plurality of files and each of said plurality of files is stored in a corresponding one of a plurality of sets of clusters, said plurality of sets of clusters being contained in said plurality of clusters, said method comprising:
- providing a first module which is designed to determine a plurality of identifiers corresponding to a specified file and store said plurality of identifiers according to a convention, wherein said plurality of identifiers specify a set of clusters corresponding to said specified file, said set of clusters being contained in said plurality of sets of clusters;
  - providing a second module which is to perform an operation on a file of interest, wherein said second module is designed to determine a desired cluster by using said plurality of identifiers stored according to said convention;
  - executing said first module while specifying said file of interest as said specified file such that a plurality of identifiers corresponding to said file of interest are

stored according to said convention; and  
executing said second module after executing said first  
module,  
wherein both of said first module and said second mod-  
ule are executed using at least some of the same loca-  
tions of said small memory.

[c8] 8. The method of claim 7, wherein said second module is  
overlaid on the same memory space on which said first  
module is loaded during execution.

[c9] 9. The method of claim 8, wherein said convention com-  
prises storing said plurality of identifiers at a pre-  
specified portion of RAM.

[c10] 10. The method of claim 9, wherein each of said plurality  
of files stores data representing a corresponding song.

[c11] 11. A machine readable medium carrying one or more  
sequences of instructions for causing a digital process-  
ing system to access data contained in a first file,  
wherein said first file is comprised in a plurality of files  
stored on a secondary storage, said secondary storage  
comprising a plurality of clusters, a file allocation table  
(FAT) indicating a corresponding set of clusters allocated  
to each of said plurality of files, wherein execution of  
said one or more sequences of instructions by one or

more processors contained in said digital processing system causes said one or more processors to perform the actions of:

determining a set of identifiers, wherein each of said set of identifiers identifies a corresponding one of said set of clusters allocated to said first file; and

storing said set of identifiers in a random access memory (RAM),

wherein storing said set of identifiers in said RAM enables faster access to data related to said first file, and not having to store said entire FAT in said RAM minimizes memory space requirements in said RAM.

[c12] 12. The machine readable medium of claim 11, wherein each of said plurality of clusters is identified by a corresponding one of a plurality of identifiers, said FAT storing each of said sets of identifiers in the form of a linked list, wherein an order specified by said linked list indicates the sequence in which said set of clusters are used to store data contained in said first file, further comprising:

traversing said linked list to retrieve said set of identifiers in said order, wherein said storing stores said set of identifiers in said RAM.

[c13] 13. The machine readable medium of claim 12, wherein said set of identifiers are stored according to a technique

by which each of said set of identifiers can be retrieved with fewer instructions than the number of instructions required to access the same identifier from said FAT in said secondary storage.

[c14] 14. The machine readable medium of claim 12, wherein said set of identifiers are stored in the form of an array from which each identifier can be retrieved by a single access.

[c15] 15. The machine readable medium of claim 14, wherein said array comprises an associative array.

[c16] 16. The machine readable medium of claim 14, further comprising:  
receiving a start offset of data to be accessed;  
computing a cluster index by dividing said start offset by a number of bytes in each of said plurality of clusters;  
and  
accessing said array using said cluster index to determine a specific one of said set of identifiers, wherein said data to be accessed is present in a cluster identified by said specific one of said set of identifiers.

[c17] 17. A machine readable medium carrying one or more sequences of instructions for causing a digital processing system to implement an application using a small

memory space, wherein said digital processing system supports a file system on a secondary storage, wherein said secondary storage comprises a plurality of clusters, wherein said file system comprises a plurality of files and each of said plurality of files is stored in a corresponding one of a plurality of sets of clusters, said plurality of sets of clusters being contained in said plurality of clusters, wherein execution of said one or more sequences of instructions by one or more processors contained in said digital processing system causes said one or more processors to perform the actions of:

providing a first module which is designed to determine a plurality of identifiers corresponding to a specified file and store said plurality of identifiers according to a convention, wherein said plurality of identifiers specify a set of clusters corresponding to said specified file, said set of clusters being contained in said plurality of sets of clusters;

providing a second module which is to perform an operation on a file of interest, wherein said second module is designed to determine a desired cluster by using said plurality of identifiers stored according to said convention;

executing said first module while specifying said file of interest as said specified file such that a plurality of identifiers corresponding to said file of interest are

stored according to said convention; and  
executing said second module after executing said first  
module,  
wherein both of said first module and said second mod-  
ule are executed using at least some of the same loca-  
tions of said small memory.

[c18] 18. The machine readable medium of claim 17, wherein  
said second module is overlaid on the same memory  
space on which said first module is loaded during exe-  
cution.

[c19] 19. The machine readable medium of claim 18, wherein  
said convention comprises storing said plurality of iden-  
tifiers at a pre-specified portion of a RAM.

[c20] 20. An apparatus accessing data contained in a first file,  
wherein said first file is comprised in a plurality of files  
stored on a secondary storage, said secondary storage  
comprising a plurality of clusters, a file allocation table  
(FAT) indicating a corresponding set of clusters allocated  
to each of said plurality of files, said apparatus compris-  
ing:

means for determining a set of identifiers, wherein each  
of said set of identifiers identifies a corresponding one  
of said set of clusters allocated to said first file; and  
means for storing said set of identifiers in a random ac-



cess memory (RAM),  
wherein storing said set of identifiers in said RAM enables faster access to data related to said first file, and not having to store said entire FAT in said RAM minimizes memory space requirements in said RAM.

[c21] 21. The apparatus of claim 20, wherein each of said plurality of clusters is identified by a corresponding one of a plurality of identifiers, said FAT storing each of said sets of identifiers in the form of a linked list, wherein an order specified by said linked list indicates the sequence in which said set of clusters are used to store data contained in said first file, said apparatus further comprising:

means for traversing said linked list to retrieve said set of identifiers in said order, wherein said storing stores said set of identifiers in said RAM.

[c22] 22. The apparatus of claim 21, wherein said set of identifiers are stored according to a technique by which each of said set of identifiers can be retrieved with fewer instructions than the number of instructions required to access the same identifier from said FAT in said secondary storage.

[c23] 23. The apparatus of claim 21, wherein said set of identifiers are stored in the form of an array from which each

identifier can be retrieved by a single access.

[c24] 24. The apparatus of claim 23, wherein said array comprises an associative array.

[c25] 25. The apparatus of claim 24, further comprising:  
means for receiving a start offset of data to be accessed;  
means for computing a cluster index by dividing said start offset by a number of bytes in each of said plurality of clusters; and  
means for accessing said array using said cluster index to determine a specific one of said set of identifiers, wherein said data to be accessed is present in a cluster identified by said specific one of said set of identifiers.

[c26] 26. An apparatus implementing an application in a system containing a small memory, wherein said system supports a file system on a secondary storage, wherein said secondary storage comprises a plurality of clusters, wherein said file system comprises a plurality of files and each of said plurality of files is stored in a corresponding one of a plurality of sets of clusters, said plurality of sets of clusters being contained in said plurality of clusters, said apparatus comprising:  
providing a first module which is designed to determine a plurality of identifiers corresponding to a specified file and store said plurality of identifiers according to a con-

vention, wherein said plurality of identifiers specify a set of clusters corresponding to said specified file, said set of clusters being contained in said plurality of sets of clusters;

providing a second module which is to perform an operation on a file of interest, wherein said second module is designed to determine a desired cluster by using said plurality of identifiers stored according to said convention;

executing said first module while specifying said file of interest as said specified file such that a plurality of identifiers corresponding to the clusters of said file of interest are stored according to said convention; and executing said second module after executing said first module,

wherein both of said first module and said second module are executed using at least some of the same locations of said small memory.

[c27] 27. The apparatus of claim 26, wherein said second module is overlaid on the same memory space on which said first module is loaded during execution.

[c28] 28. The apparatus of claim 27, wherein said convention comprises storing said plurality of identifiers at a pre-specified portion of a RAM.